

AD-A075 236

DEFENCE RESEARCH ESTABLISHMENT VALCARTIER (QUEBEC)
EXTRACTEURS SEQUENTIELS POUR L'ACQUISITION DE CIBLES SUR IMAGES--ETC(U)
AUG 79 L SEVIGN
DREV-R-4153/79

F/6 5/8

UNCLASSIFIED

NL

OF
AD A
075236



NON CLASSIFIÉ

UNLIMITED
DISTRIBUTION
ILLIMITÉE

CRDV RAPPORT 4153/79

DOSSIER: 3621J-007

AOUT 1979

DREV REPORT 4153/79

FILE: 3621J-007

AUGUST 1979

B-53

AD A 075236

EXTRACTEURS SÉQUENTIELS POUR
L'ACQUISITION DE CIBLES SUR IMAGES

L. Sévigny

DDC
RECEIVED
OCT 22 1979
A

DDC FILE COPY

Centre de Recherches pour la Défense
Defence Research Establishment
Valcartier, Québec

BUREAU - RECHERCHE ET DÉVELOPPEMENT
MINISTÈRE DE LA DÉFENSE NATIONALE
CANADA

RESEARCH AND DEVELOPMENT BRANCH
DEPARTMENT OF NATIONAL DEFENCE
CANADA

UNCLASSIFIED

79 10 19 046

CRDV R-4153/79
DOSSIER: 3621J-007

UNCLASSIFIED

14 DREV- R-4153/79
FILE: 3621J-007

6

EXTRACTEURS SEQUENTIELS POUR
L'ACQUISITION DE CIBLES SUR IMAGES

(Sequential extractors for target acquisition on images)

10

L./Sévigny

11 / Aug 79

12 61

CENTRE DE RECHERCHES POUR LA DEFENSE

✓ DEFENCE RESEARCH ESTABLISHMENT

VALCARTIER

Tel. (418) 844-4271

Québec, Canada

August/aout 1979

NON CLASSIFIE

404 945

JOB

NON CLASSIFIE

i

RESUME

Ce rapport décrit 2 extracteurs séquentiels d'usage général bien qu'ils aient été développés en fonction d'une application bien particulière, à savoir l'acquisition de cibles sur images. L'extraction des traits caractéristiques implique une opération préalable visant à délimiter les composantes connexes, ou objets, représentés sur une image. Les 2 extracteurs réalisent cette opération par étiquetage et propagation; l'un, des points de contours de l'objet et l'autre des pixels constituant eux-mêmes. Dans le premier cas, nous démontrons qu'il suffit de 4 inégalités simples, et non de 8 comme dans l'article de Agrawala et Kulkarni, pour identifier complètement le contour d'un objet. Le second procédé, par contre, requiert un nombre de comparaisons qui est fonction du nombre de voisins immédiats que l'on associe à un point de l'image. Toutefois, aucun des 2 extracteurs ne semble posséder, globalement, un avantage écrasant sur l'autre. (NC)

ABSTRACT

This report describes 2 general-purpose sequential extractors that were primarily developed for a specific application, namely target acquisition on images. The process of extracting features implies an initial operation aimed at delimitating the connected components, or objects, represented on an image. Both extractors realize this operation by labeling and propagating, for the first extractor, the boundary points and for the second one, the pixels themselves. In the first case, we show that one needs only 4 simple inequalities, instead of 8 as in the article by Agrawala and Kulkarni, to fully identify the boundaries of an object. The second process, on the contrary, needs a number of comparisons that is a function of the number of nearest neighbors one assigns to an image point. However, on the whole, none of the 2 extractors seem to possess an overwhelming advantage over the other. (U)

Accession For	
NTIS GRA&I	<input checked="checked" type="checkbox"/>
DDC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By _____	
Distribution/ _____	
Availability Codes	
Dist.	Avail and/or special
A	

NON CLASSIFIE

ii

TABLE DES MATIERES

	RESUME/ABSTRACT	i
1.0	INTRODUCTION	1
2.0	CONSIDERATIONS GENERALES RELATIVES A L'ACQUISITION DE CIBLES SUR IMAGES	5
3.0	EXTRACTEURS SEQUENTIELS POUR L'ACQUISITION DE CIBLES SUR IMAGES	10
3.1	Etiquetage et propagation des points de contours - Extracteur no. 1	12
3.2	Etiquetage et propagation des pixels individuels - Extracteur no. 2	28
4.0	EVIDENCES EXPERIMENTALES	35
5.0	CONCLUSION	51
6.0	REFERENCES	53
	FIGURES 1 à 16	
	TABLEAUX I et II	
	APPENDICE A - Organigramme de la procédure d'identification des points de contours	54

NON CLASSIFIE

1

1.0 INTRODUCTION

Nous nous intéressons depuis quelques années (réf. 1-3) à la détection de cibles sur images fournies par des caméras infrarouge. A cet effet, nous avons conçu et simulé en ordinateur un système hypothétique (système AIRTAS) d'acquisition automatique de cibles infrarouges. Le processus d'acquisition comprend plus que la simple détection. En fait, celle-ci n'en constitue que la première étape, les deux suivantes étant la classification et l'identification de la cible. Si tous nos efforts ont été concentrés jusqu'ici sur la détection proprement dite, c'est que cette première étape est primordiale. Quelle que soit son efficacité au niveau de la classification et de l'identification, un système d'acquisition n'est d'aucune utilité si son taux d'échec à la détection est très élevé. Par ailleurs, comme l'indique le critère de Johnson, pour franchir avec succès toutes les étapes du processus d'acquisition, la cible doit, au départ, avoir une certaine taille minimale. Une cible plus petite pourra possiblement être classée, mais sera difficilement identifiable, sinon pas du tout. Enfin, pour ce qui est des cibles ponctuelles, le processus d'acquisition s'arrête obligatoirement après la première étape. Ce fait pourrait même être utilisé comme définition pratique d'une cible ponctuelle. En définitive, le processus d'acquisition dépend énormément de la taille des cibles et, pour en retirer le maximum, il est préférable de les regrouper, à tout le moins, en 2 grandes classes: cibles ponctuelles et cibles étendues.

Les réf. 2 et 3 décrivent les deux premières versions du système AIRTAS. Les caractéristiques escomptées de ce système, que ne possèdent que partiellement les deux premières versions, sont:

- a) fonctionnement en temps réel,
- b) automatique,
- c) efficace, c'est-à-dire une très bonne probabilité de détection, tout en ayant un taux de fausses alarmes relativement faible,
- d) réalisable pratiquement à peu de frais,
- e) d'un encombrement minimum.

Il s'agit évidemment là de caractéristiques éminemment souhaitables mais, sans doute, grandement idéalisées étant donné l'état actuel des connaissances dans ce domaine. Quoi qu'il en soit, on ne peut espérer obtenir de telles caractéristiques sans restreindre considérablement les algorithmes. Ainsi, un fonctionnement en temps réel implique un système adapté au mode d'opération et au rythme du capteur, c'est-à-dire, en général, une image débitée de façon séquentielle à une cadence de 30 images/seconde. Le capteur dicte, de plus, le format de l'image lequel peut, en soi, éliminer certaines méthodes de traitement; celles qui nécessitent une transformation de Fourier, par exemple. Considérons, à titre d'exemple, une image thermoscopique comportant, comme celle qui se trouve un peu plus loin dans ce rapport, 420 lignes de 335 pixels. Pour pouvoir traiter de telles images en temps réel, le système doit être en mesure d'examiner plus de 4 millions de pixels par seconde. C'est une quantité impressionnante de données qu'il faut prendre garde d'amplifier par des algorithmes mal conçus. Ainsi, le processus d'acquisition doit

préférentiellement se faire en un seul balayage de l'image. Autrement, il faudrait, toutes autres considérations mises à part, pourvoir le système d'une mémoire image ce qui en augmenterait à la fois le coût et l'encombrement. Par contre, si les applications projetées peuvent tolérer une cadence plus lente que 30 images/s, on peut envisager, comme solution de rechange au balayage multiple, d'étendre le processus d'acquisition sur plusieurs images consécutives. Il va sans dire que, dans ces conditions, les algorithmes doivent pouvoir s'accommoder d'images qui ne sont pas rigoureusement identiques.

Des considérations précédentes il ressort que, pour concevoir un système d'acquisition automatique de cibles répondant à nos attentes, les algorithmes de traitement doivent posséder les caractéristiques générales suivantes:

- 1) séquentiels, c'est-à-dire que le traitement porte non pas sur l'image entière mais sur des pixels individuels ou, à la rigueur, sur une ou plusieurs lignes de l'image mais dans l'ordre naturel de succession de l'un ou l'autre;
- 2) balayage unique, pour des raisons de vitesse aussi bien que de coût et d'encombrement;
- 3) simples, en ce sens que le nombre moyen d'opérations par pixel doit être petit étant donné que le nombre de pixels à traiter par seconde est très grand;

- 4) universels, c'est-à-dire que les algorithmes doivent pouvoir s'appliquer à des images montrant des scènes présentant une grande diversité de fonds, sous des éclairagements tout aussi variés. Les algorithmes ne doivent donc pas être restreints à un type particulier d'image. La meilleure façon d'atteindre ce résultat consiste sans doute à concevoir des algorithmes qui s'adaptent automatiquement à l'image traitée.

Ces caractéristiques n'ont toutefois rien d'absolu. Il pourra être nécessaire de les redéfinir pour tenir compte des dernières trouvailles de la technologie.

La réf. 3 aborde le problème de l'acquisition automatique de cibles sur images infrarouge d'une façon que l'on pourrait qualifier de classique. La procédure comprend quatre étapes:

- a) la segmentation de l'image,
- b) l'identification des composantes connexes, ou objets, de l'image segmentée,
- c) l'extraction des traits caractéristiques des objets,
- d) la classification des objets.

La première étape doit forcément être adaptée à la nature de l'image traitée et à celle des objets que l'on veut en extraire. De même, les

étapes 3 et 4 sont directement fonction des applications considérées. Par contre, la deuxième étape est une procédure tout à fait générale qui n'est nullement liée à une application particulière. C'est de celle-ci dont il est question dans le présent rapport.

Ce travail a été effectué au CRDV entre mai 1977 et avril 1978, dans le cadre du NCP 21J07 "Target Acquisition".

2.0 CONSIDERATIONS GENERALES RELATIVES A L'ACQUISITION DE CIBLES SUR IMAGES

La fig. 1 schématise un système d'acquisition de cibles sur images. Un tel système comporte obligatoirement les trois unités suivantes: un discriminateur, un extracteur et un classifieur. Le discriminateur peut être précédé ou non d'une unité de prétraitement destinée, par exemple, à améliorer le rapport signal/bruit de l'image ou, encore, à en accuser certains traits dans le but d'accroître l'efficacité du discriminateur. Ce dernier a pour fonction de dissocier du fond de la scène tous les objets présentant des caractéristiques qui s'apparentent à celles des cibles recherchées. Il existe plus d'une façon de segmenter une image donnée. Chacune d'elles doit d'abord être évaluée en fonction de sa probabilité de détection, c'est-à-dire, parmi tous les objets extraits de l'image, retrouve-t-on toutes les cibles réellement présentes? Il faut ensuite considérer le taux de fausses alarmes. Ce problème est un peu plus complexe car il faut tenir compte du classifieur. Pour ne pas déborder du cadre de ce rapport, contentons nous de dire que de 2 algorithmes de segmentation possédant la même

NON CLASSIFIE
6

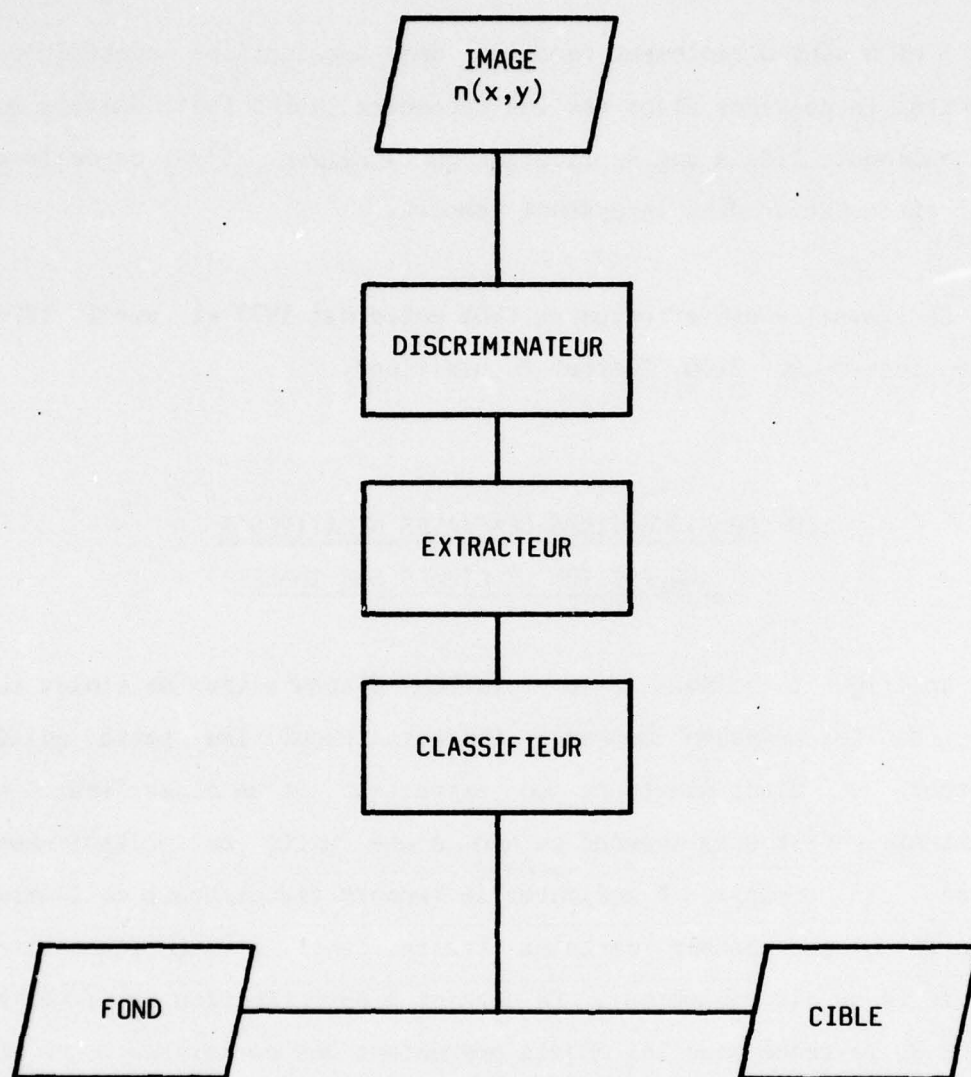


FIGURE 1 - Schéma d'un système d'acquisition de cibles sur images

probabilité de détection, on choisira celui qui exige le moins du classifieur.

Dans le cas d'une image infrarouge, on peut fonder la procédure de segmentation sur le fait que la cible est généralement plus chaude que les objets environnants. La fig. 2 montre 3 segmentations différentes d'une image thermoscopique (fig. 2a) comportant 420 lignes de 335 pixels et où les objets chauds sont représentés en blanc. Désignons par $n(x,y)$ la valeur du niveau de gris de l'image au point (x,y) et représentons la fonction seuil (fig. 2b) par $s(y)$. L'image segmentée (fig. 2c) est alors obtenue de la façon suivante:

$$\text{seg}(x,y) = \begin{cases} 0 & \text{si } n(x,y) \leq s(y) \\ 1 & \text{si } n(x,y) > s(y) \end{cases} \quad [1]$$

La première fonction seuil, fig. 2b(1), a une valeur constante pour toutes les lignes de l'image: $s(y) = \text{constante} = n_f$. Il s'agit, en fait, du générateur de silhouettes des réf. 2 et 3. On remarquera qu'on ne retrouve sur l'image de la fig. 2c(1) qu'une seule des 3 cibles, soit la plus brillante et en l'occurrence le blindé. Cet exemple montre que la probabilité de détection du générateur de silhouettes laisse parfois à désirer. Les fonctions seuil des fig. 2b(2) et 2b(3) sont produites par des variantes, que nous décrirons dans un prochain rapport, de la précédente procédure de segmentation. Les deux possèdent une excellente probabilité de détection.

Ainsi que le montre la fig. 2c, l'image segmentée contient généralement, outre un certain nombre de cibles, plusieurs objets parasites que le classifieur a pour rôle d'éliminer. Mais pour cela il

NON CLASSIFIE
8

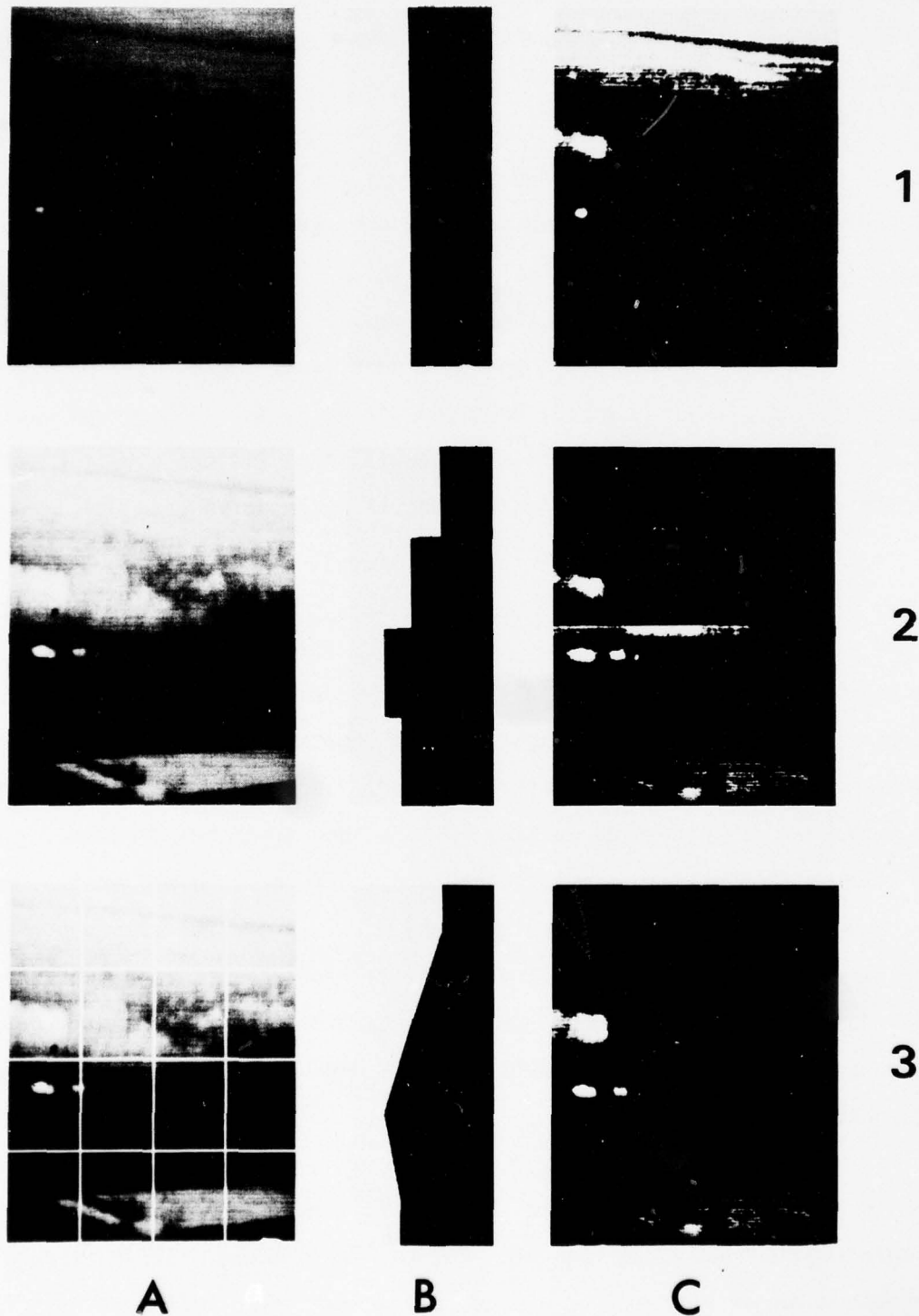


FIGURE 2 - Différentes façons de segmenter une image thermoscopique:
a) image originale, 420 x 335 pixels;
b) fonctions seuil;
c) images segmentées.

faut d'abord dénombrer et délimiter les objets présents, et ensuite les réduire à certains traits caractéristiques qui permettront de les départager. Ces deux opérations sont exécutées parallèlement par l'extracteur. Il ne saurait en être autrement si l'on veut, comme nous l'avons indiqué dans l'introduction, que tout le processus d'acquisition ne nécessite guère plus d'un seul balayage de l'image. Par ailleurs, tel qu'indiqué également dans l'introduction, il est préférable que le processus se déroule de façon séquentielle. Il va sans dire que cette dernière restriction impose que la procédure de segmentation soit elle-même séquentielle. De plus, on doit se limiter à des traits caractéristiques qu'il est possible d'extraire de la même façon.

L'opération du classifieur n'implique aucun balayage de l'image puisqu'elle porte uniquement sur les traits caractéristiques des objets extraits de l'image. Cette étape est par ailleurs facilement réalisable en temps réel puisque le nombre d'objets à classer est généralement petit et qu'il en est de même du nombre de traits caractéristiques.

3.0 EXTRACTEURS SEQUENTIELS POUR L'ACQUISITION DE CIBLES SUR IMAGES

Dans un système d'acquisition de cibles sur images, l'opération d'extraction des traits caractéristiques des objets produits par la segmentation est confiée à une unité appelée extracteur. Nous supposerons ici que la segmentation produit non pas une image segmentée comme telle, ce qui impliquerait que le système possède une mémoire image, mais une fonction seuil qui, tel qu'illustré dans la section précédente, nous permet de déterminer si un pixel quelconque de l'image appartient à une cible potentielle ou au fond de la scène. Etant donné une telle fonction, nous verrons qu'il suffit de balayer l'image une seule fois, de façon séquentielle, pour mener à terme tout le processus d'acquisition.

L'extracteur no. 1 décrit dans la section 3.1 est une variante de celui exposé dans l'article de Agrawala et Kulgarni (réf. 4). Dans les réf. 2 et 3, par ailleurs, nous avons décrit un autre type d'extracteur que nous avons désigné sous le nom de synthétiseur de silhouettes. Dans la section 3.2 de ce rapport, nous en reprenons succinctement la description dans les mêmes termes que le premier de manière à mieux faire ressortir les ressemblances et les différences que présentent ces 2 extracteurs. Mentionnons immédiatement que l'un et l'autre possèdent plusieurs particularités éminemment souhaitables dans la perspective d'une réalisation pratique de tels appareils:

- 1) ils fonctionnent de façon purement séquentielle,

- 2) il leur suffit d'un seul balayage de l'image pour extraire toutes les données nécessaires à la détermination des traits caractéristiques,
- 3) ils n'altèrent pas l'image,
- 4) il n'est pas nécessaire d'enregistrer préalablement l'image.

Ces particularités font que ces extracteurs peuvent être intégrés à des systèmes en temps réel, pourvus ou non d'une mémoire image, ce qui est très important. L'addition d'une mémoire image à un système qui pourrait très bien s'en passer augmente le coût du système et son encombrement. Ce dernier facteur a une grande importance dans certaines applications, comme par exemple l'autoguidage de petits missiles.

Les extracteurs dont il est ici question sont des unités logicielles, désignées respectivement ESTRAC et SSIL, simulant des appareils hypothétiques du même nom. Décrire l'un c'est donc décrire l'autre et les mots pour le dire évoqueront tantôt l'un, tantôt l'autre. Les 2 extracteurs emploient un procédé analogue pour définir les objets, soit l'étiquetage et la propagation des points de contours (ESTRAC), ou des pixels eux-mêmes (SSIL).

3.1 Etiquetage et propagation des points de contours - Extracteur no. 1

3.1.1 Description de l'extracteur

Cet extracteur peut traiter des images numériques de taille quelconque sans qu'il faille modifier la capacité de sa mémoire. Celle-ci, en effet, n'est pratiquement pas fonction des dimensions de l'image et est plutôt fixée par le nombre de traits caractéristiques que l'on désire extraire de l'image, par la nécessité de conserver certains résultats intermédiaires et, dans une moindre mesure, par le nombre maximum d'objets qu'il est possible de trouver sur une même image. En fait la mémoire de l'extracteur doit être suffisamment grande pour accommoder le tableau de la fig. 3, ainsi qu'un second tableau, fig. 4, fonction des applications. Le premier sert à l'identification des points de contours et comprend 8 rubriques ou colonnes réparties sur 2 volets qui sont le pendant l'un de l'autre. Le second tableau est destiné à recueillir, en cours de balayage, toutes les données nécessaires à la détermination des traits caractéristiques. Il doit comporter au moins autant de colonnes qu'il y a de traits caractéristiques à déterminer, mais possiblement plus s'il est nécessaire, comme ici, de conserver des résultats intermédiaires.

3.1.1.1 Tableau des traits caractéristiques (t.t.c.)

Ainsi que nous l'avons mentionné, le tableau des traits caractéristiques, ou t.t.c., est fonction des applications. Dans le cas qui nous occupe, à savoir l'acquisition de cibles à partir d'images infrarouge, il est nécessaire de prévoir 12 colonnes (fig. 4).

```
1: point de contour gauche
2: point de contour droit
```

FIGURE 3 - Le tableau d'identification des contours (t.i.c.) comporte 8 colonnes et un nombre de lignes limité arbitrairement à 64. La moitié droite du tableau est le pendant de la moitié gauche. La première se rapporte à la ligne courante et la seconde à la ligne qui précède.

1: coin supérieur gauche
2: centre thermique

The diagram shows a scene within a dashed rectangular frame. A vertical arrow on the left points downwards, indicating a direction of view. The scene contains several regions: a large, irregularly shaped region filled with diagonal hatching, and several smaller, irregularly shaped regions that are white (unfilled). The hatched region is labeled with p_1 , p_2 , p_3 , p_4 , and p_5 . The white regions are labeled with t_1 and t_2 . The corners of the dashed frame are labeled with b_1 , b_2 , b_3 , b_4 , and b_5 . The top-left corner is labeled $c.s.g.$ and the bottom-right corner is labeled $c.i.d.$. A legend on the left shows a hatched square labeled 'objet' and a white square labeled 'trou ou point creux'.

FIGURE 5 - Cette figure illustre la terminologie des composantes connexes. Les t_i désignent des trous, les p_i des points creux et les b_i des bras. L'extracteur ne voit que les points creux p_1 et p_2 , c'est-à-dire les points creux pénétrant dans la forme en sens inverse du balayage. La figure en pointillé est le rectangle minimum contenant la forme.

L'affectation assignée aux différentes colonnes de la fig. 4 n'est pas nécessairement celle qui a cours lors du balayage de l'image. Il en est ainsi, d'abord parce qu'il faut amender le t.t.c. chaque fois que l'extracteur rencontre un nouveau point de contour et, ensuite, parce qu'on doit, à l'occasion, lorsque le tableau est quasiment rempli, remanier complètement le t.t.c., en fusionnant les objets contigus et en éliminant les objets aberrants, pour être en mesure de poursuivre le processus d'extraction. Les 4 premières colonnes du t.t.c. renferment, en cours de balayage, les coordonnées du coin supérieur gauche (c.s.g.) et celles du coin inférieur droit (c.i.d.), du rectangle minimum contenant l'objet (fig. 5). Une fois le balayage complété, l'abscisse et l'ordonnée du c.i.d. sont remplacées respectivement par la largeur hors-tout de l'objet:

$$\ell = x_{c.i.d.} - x_{c.s.g.} \quad [2]$$

et par sa hauteur hors-tout:

$$h = y_{c.i.d.} - y_{c.s.g.} \quad [3]$$

Ces quantités définissent le rapport ℓ/h qui entre dans la détermination du pointage attribué à chacun des objets (réf. 3). L'étendue de l'objet, c'est-à-dire le nombre total de points, N , constituant l'objet, est donnée dans la cinquième colonne du t.t.c.. Durant le balayage, ce nombre peut être négatif. Il s'agit d'un artifice qui permet à l'extracteur de faire la différence entre l'objet lui-même, et un trou ou un point creux (fig. 5) de l'objet. Cet artifice permet, en outre, de fusionner sans discernement des objets et des trous, tout en étant

assuré que l'étendue de l'objet résultant sera exacte. Les colonnes 6, 7, 8, 9, 11 et 12 du t.t.c. accumulent, pour chacun des objets que l'extracteur décèle, les quantités suivantes:

$$\sum \sum n(x,y) = m_{0,0} \quad [4]$$

$$\sum \sum x n(x,y) = m_{1,0} \quad [5]$$

$$\sum \sum y n(x,y) = m_{0,1} \quad [6]$$

$$\sum \sum x y n(x,y) = m_{1,1} \quad [7]$$

$$\sum \sum x^2 n(x,y) = m_{2,0} \quad [8]$$

$$\sum \sum y^2 n(x,y) = m_{0,2} \quad [9]$$

où $n(x,y)$ désigne le niveau de gris de l'objet au point (x,y) . Une fois le balayage complété et le t.t.c. mis à jour, on déduit facilement à partir de ces quantités:

- a) le niveau de gris moyen, \bar{n} , qui est ici fonction de la température moyenne de l'objet:

$$\bar{n} = m_{0,0}/N; \quad [10]$$

- b) les coordonnées de l'équivalent du centre de gravité de l'objet, que nous appellerons centre thermique:

$$\bar{x} = m_{1,0}/m_{0,0}, \quad [11]$$

$$\bar{y} = m_{0,1}/m_{0,0};$$

- c) l'équivalent de l'axe principal d'inertie de l'objet, que nous appellerons axe thermique:

$$\theta = \frac{\arctan (2M_{1,1}/(M_{2,0}-M_{0,2})),}{2} \quad [12]$$

où

$$M_{1,1} = m_{1,1} - \bar{x}\bar{y}m_{0,0}, \quad [12a]$$

$$M_{2,0} = m_{2,0} - \bar{x}\bar{x}m_{0,0}, \quad [12b]$$

$$M_{0,2} = m_{0,2} - \bar{y}\bar{y}m_{0,0}; \quad [12c]$$

- d) les rayons d'inertie de l'objet:

$$r_1 = m_{0,2} + m_{1,1}/\tan\theta, \quad [13]$$

$$r_2 = m_{2,0} + m_{1,1}/\tan\theta,$$

qui mesurent en quelque sorte l'envergure de l'objet respectivement dans la direction de l'axe principal d'inertie et dans une direction perpendiculaire à ce dernier;

e) l'allongement de l'objet:

$$a = r_1/r_2, \quad [14]$$

lequel, contrairement au rapport ℓ/h , est indépendant de l'orientation de l'objet;

f) l'encombrement de l'objet ou la proportion du rectangle circonscrit occupée par l'objet:

$$e = N/(\ell x h). \quad [15]$$

Nous avons décrit jusqu'ici toutes les colonnes du t.t.c. à l'exception de la dixième. Celle-ci est en quelque sorte le pivot du t.t.c., et en définitive de l'extracteur lui-même. Il s'agit d'une table de substitution qui associe tel objet ou tel trou avec tel autre objet ou tel autre trou. C'est sur la base des indications fournies par la table de substitution que se font les fusions. Les paragraphes qui suivent feront mieux voir le rôle de cette table dans le processus d'extraction des traits caractéristiques.

3.1.1.2 Tableau d'identification des contours (t.i.c.)

Tout ce dont l'extracteur a besoin pour démarrer, c'est le niveau de gris maximal de l'arrière-plan, n_f , lequel lui est fourni par la fonction seuil (voir section 2.0). Désignons par $n(x,y)$ la valeur du niveau de gris de l'image au point (x,y) . Les points de l'image pour lesquels $n(x,y) > n_f$ sont naturellement regroupés en un certain nombre

d'objets isolés les uns des autres, c'est-à-dire séparés par au moins un pixel dont le niveau de gris est plus petit ou égal à n_f . La fonction ultime de l'extracteur consiste à déterminer les traits caractéristiques des différents objets ainsi définis. Pour y parvenir, il doit d'abord identifier et délimiter lesdits objets.

Au départ, les 4 premières colonnes du tableau d'identification des contours (t.i.c.), sont remplies avec des zéros. Il en est de même de toutes les colonnes du t.t.c., à l'exception de la dixième, la table de substitution, qui est remplie avec des nombres allant de 1 jusqu'au nombre total de lignes du tableau. L'extracteur balaye l'image point par point, de gauche à droite et de haut en bas, faisant une pause à la fin de chaque ligne. Cette opération a pour but de localiser les points de contours. On fait une distinction entre un point de contour gauche (p.c.g.) et un point de contour droit (p.c.d.). Un p.c.g. est un point d'image plus grand que n_f , qui est encadré à gauche par un point plus petit ou égal à n_f , et à droite par un point également plus grand que n_f . C'est la situation inverse qui prévaut dans le cas d'un p.c.d. L'abscisse d'un p.c.g. est désignée par x_L , et celle d'un p.c.d. par x_R . Par définition, x_L et x_R délimitent un segment dont tous les points sont plus grands que n_f . Au fur et à mesure que l'extracteur rencontre un point de contour, il détermine son abscisse et entre le résultat dans la colonne 5 ou la colonne 6 du tableau de la fig. 1, dépendant de la nature du point de contour. Une fois qu'une ligne complète de l'image a été examinée, le balayage est momentanément interrompu afin de permettre à l'extracteur d'identifier les segments rencontrés. Cette opération consiste à remplir les colonnes 7 et 8 du t.i.c. (fig. 3) à partir des données contenues dans les 4 premières colonnes, à savoir les abscisses

des points de contours des segments de la ligne précédente et leur numéro d'identification. Ainsi, la moitié gauche du t.i.c. concerne les segments précédemment identifiés de la ligne k-1, et la moitié droite les segments à identifier de la ligne k.

3.1.2 Identification des points de contours

Le processus d'identification des points de contours postule que ces derniers conservent leur identité tout en se propageant d'une ligne à l'autre de l'image. Autrement dit, un p.c.g. (p.c.d.) de la ligne k-1 ne peut être relié qu'à un p.c.g. (p.c.d.) de la ligne k. D'après Agrawala et Kulgarni (réf. 4), un p.c.g. de la ligne k est le prolongement d'un p.c.g. de la ligne k-1 si:

$$x_{Rj-1}^{(k-1)} < x_{Li}^{(k)} \leq x_{Rj}^{(k-1)}, \quad [16]$$

$$x_{Ri-1}^{(k)} < x_{Lj}^{(k-1)} \leq x_{Ri}^{(k)}. \quad [17]$$

De même, un p.c.d. de la ligne k est le prolongement d'un p.c.d. de la ligne k-1 si:

$$x_{Li}^{(k)} \leq x_{Rj}^{(k-1)} < x_{Li+1}^{(k)}, \quad [18]$$

$$x_{Lj}^{(k-1)} \leq x_{Ri}^{(k)} < x_{Lj+1}^{(k-1)}. \quad [19]$$

(Il est à noter que notre définition d'un p.c.d. diffère de celle utilisée par Agrawala et Kulgarni (réf. 4). En fait, leur p.c.d. correspond au point d'image qui suit immédiatement notre p.c.d.). On peut écrire ces 4 inégalités doubles comme 8 inégalités simples:

$$x_{Li}(k) \leq x_{Rj}(k-1), \quad [20]$$

$$x_{Li}(k) > x_{Rj-1}(k-1), \quad [21]$$

$$x_{Lj}(k-1) \leq x_{Ri}(k), \quad [22]$$

$$x_{Lj}(k-1) > x_{Ri-1}(k), \quad [23]$$

$$x_{Li}(k) \leq x_{Rj}(k-1), \quad [24]$$

$$x_{Rj}(k-1) < x_{Li+1}(k), \quad [25]$$

$$x_{Lj}(k-1) \leq x_{Ri}(k), \quad [26]$$

$$x_{Ri}(k) < x_{Lj+1}(k-1). \quad [27]$$

Certaines de ces inégalités sont identiques; d'autres sont redondantes à la condition d'identifier les segments de la ligne k de façon systématique, c'est-à-dire en les prenant un par un, en allant toujours de gauche à droite, et en n'abandonnant un segment pour passer au suivant qu'une fois ses 2 points de contours bien identifiés. Ainsi, pour un segment donné, i_0 , de la ligne k (fig. 6), on recherche, parmi les segments précédemment identifiés de la ligne $k-1$, un segment, j_0 , satisfaisant l'inégalité 20. Du même coup, l'inégalité 21 est

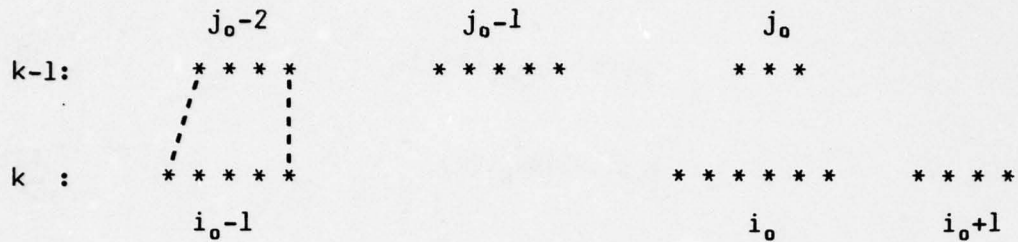


FIGURE 6 - Le segment j_0 se prolonge en i_0 tandis que les segments j_0-1 et i_0+1 marquent respectivement la fin d'un objet et le début d'un autre.

satisfaite puisque, si tel n'était pas le cas, le segment satisfaisant 20 ne serait pas j_0 mais bien j_0-1 . De plus, l'inégalité 24 est également satisfaite étant donné qu'elle est identique à 20. La recherche du segment j_0 débute avec celui qui suit immédiatement le dernier segment relié (segment j_0-2 dans l'exemple de la fig. 6) et se poursuit, vers la droite, jusqu'à épuisement de ceux de la ligne $k-1$. Ce faisant, certains segments pourront être rejetés (segment j_0-1 de la fig. 6). Ce sont autant d'objets qui disparaissent, c'est-à-dire des objets dont le contour est complètement délimité et qu'on ne retrouve pas sur la ligne k et les lignes subséquentes. Par ailleurs, il se peut qu'aucun des segments de la ligne $k-1$ ne vérifient l'inégalité 20 (c'est ce qui se produit avec le segment i_0+1 de la fig. 6). Dans ces conditions, le segment i_0 marque le début d'un nouvel objet et ses points de contours sont identifiés à l'aide d'un numéro encore inutilisé. Passons maintenant à l'inégalité 22, laquelle est identique à l'inégalité 26. Comme dans le cas de l'inégalité 20, une violation de l'inégalité 22 signifie l'apparition d'un nouvel objet (fig. 7). Par

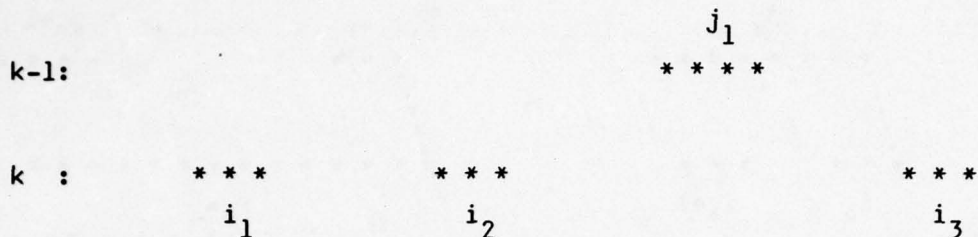


FIGURE 7 - Le segment i_3 viole l'inégalité 20 tandis que les segments i_1 et i_2 la satisfont. Toutefois, ces 2 derniers segments violent l'inégalité 22. Dans un cas comme dans l'autre, un nouvel objet apparaît.

contre, si les segments j_0 et i_0 vérifient les inégalités 20 et 22 on en conclut que le p.c.g. de i_0 est le prolongement du p.c.g. du segment j_0 ; l'inégalité 23 est ici superflue. En effet, en vertu de la procédure adoptée, nous ne sommes jamais placés dans une situation qui va à l'encontre de cette inégalité.

Une fois l'identification du p.c.g. du segment i_0 complétée, on s'attaque au p.c.d. Le fait que le p.c.g. du segment i_0 soit le prolongement du p.c.g. du segment j_0 n'implique nullement que le p.c.d. de l'un soit également le prolongement du p.c.d. de l'autre. La fig. 8 illustre 2 situations qui prouvent cette assertion. La première, (fig. 8a), viole l'inégalité 25, ce qui se traduit par l'apparition d'un trou au niveau de la ligne k. Dans le cas de la fig. 8b, c'est l'inégalité 27 qui est violée par suite de la disparition d'un trou entre les segments j_0 et j_0+1 . On traite l'apparition ou la disparition d'un trou

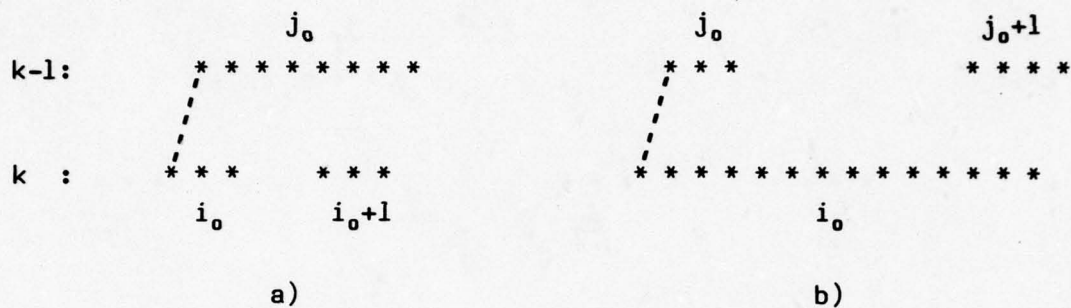


FIGURE 8 - Dans les 2 cas, le p.c.g. du segment i_0 est le prolongement de celui du segment j_0 . Toutefois, comme on peut le constater, ceci n'implique nullement que le p.c.d. de l'un soit également le prolongement du p.c.d. de l'autre.

essentiellement de la même manière qu'un objet. En résumé:

- a) le p.c.g. du segment i_0 est le prolongement du p.c.g. du segment j_0 si les inégalités 20 et 22 sont satisfaites,
- b) le p.c.d. du segment i_0 est le prolongement du p.c.d. du segment j_0 si les inégalités 25 et 27 sont satisfaites,
- c) le segment i_0 en entier est le prolongement de tout le segment j_0 si les 4 inégalités 20, 22, 25 et 27 sont satisfaites, et
- d) la non vérification de l'une ou l'autre de ces inégalités se traduit par l'apparition ou la disparition d'un objet ou d'un trou.

Il suffit donc en dernier ressort de 4 inégalités, et non de 8, comme dans la réf. 4, pour identifier les points de contours. Le lecteur trouvera dans l'appendice un organigramme détaillé, en langage FORTRAN, de la procédure d'identification des points de contours.

Lorsqu'un objet disparaît, le numéro d'identification du p.c.g. du segment terminal, N_G , est comparé au numéro d'identification de son p.c.d., N_D . Si ces 2 numéros diffèrent, on détermine lequel est le plus grand, N_D disons, et on remplace, dans la table de substitution, tous les N_D par N_G . Dans le cas d'un trou, la procédure est un peu plus compliquée étant donné qu'on doit d'abord rechercher l'objet auquel le trou appartient et, pour ce faire, il peut être nécessaire d'examiner tour à tour plusieurs des segments qui précèdent le trou. Ensuite, cependant, la procédure est la même que pour un objet.

3.1.3 Aperçu du fonctionnement global de l'extracteur

L'organigramme général de l'extracteur (fig. 9) montre les différentes étapes de traitement en vue de l'extraction des traits caractéristiques de formes quelconques définies sur une image. Il est utile de rappeler que l'extracteur peut être adapté à diverses applications, bien que nous le limitons ici à l'acquisition de cibles infrarouges.

La section 3.1.1.2 décrit comment se fait la remise à zéro du t.t.c. et du t.i.c. ainsi que la procédure suivie pour l'extraction des points de contours. Au fur et à mesure que ces derniers sont extraits, il est possible d'éliminer sur-le-champ certaines formes aberrantes.

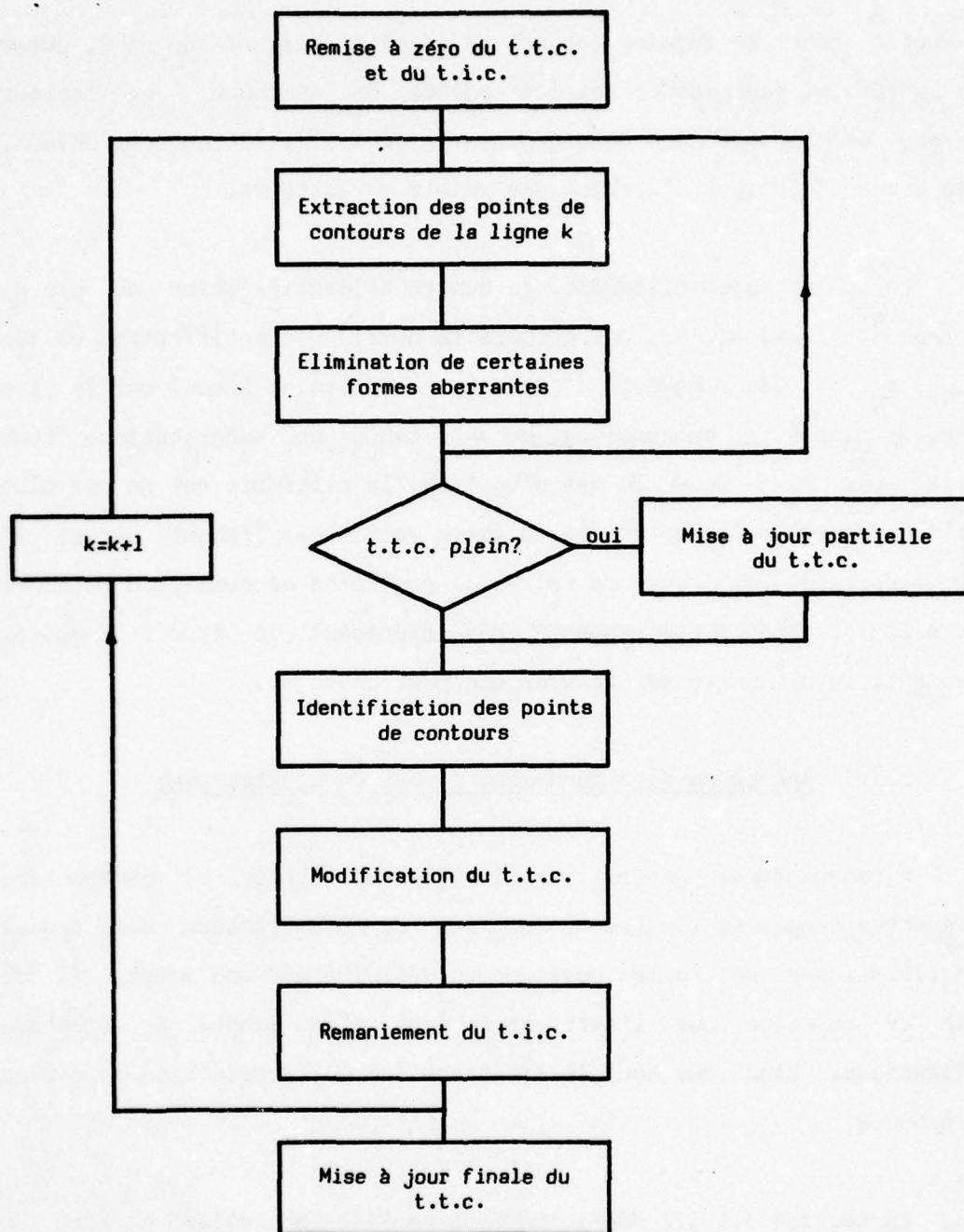


FIGURE 9 - Organigramme général de l'extracteur no. 1. Le bloc intitulé: "Identification des points de contours" est développé dans l'appendice.

Ceci contribue à accélérer le processus d'extraction, en réduisant notamment le nombre de segments à identifier. Par ailleurs, on évite ainsi de saturer la mémoire de l'extracteur, constituée des tableaux t.t.c. et t.i.c. Les objets que l'on peut immédiatement éliminer en ce point sont ceux mesurant moins de 3 pixels de large, chiffre que nous avons choisi un peu arbitrairement comme étant la limite de résolution du système mais qui ne risque guère de surestimer ladite limite. L'élimination, comme il se doit, des objets dont la hauteur mesure moins de 3 pixels est reportée à une étape ultérieure.

Avant de procéder à l'identification des points de contours, il peut être nécessaire de remanier le t.t.c. si ce dernier est sur le point d'être rempli. La meilleure façon de libérer de l'espace consiste à regrouper les objets contigus d'après les indications fournies par la table de substitution. Une autre façon réside dans l'élimination des objets aberrants suivants:

- a) ceux dont la hauteur mesure moins de 3 pixels, et
- b) ceux dont l'une des dimensions couvre plus des $3/4$ de la dimension correspondante de l'image.

Lorsque ces 2 mesures ne suffisent pas, l'extracteur doit déclarer forfait et passer à une autre image. Si la chose devait se produire trop souvent, il faudrait en conclure que la mémoire de l'extracteur n'est pas suffisamment grande pour le problème traité. Il y a ici un compromis à faire entre la capacité de la mémoire de l'extracteur, le nombre de mises à jour du t.t.c. que l'on peut tolérer, et le taux d'échecs que l'on peut se permettre.

La procédure d'identification des points de contours est décrite en détail dans la section 3.1.2. De plus, un organigramme détaillé de cette étape de traitement figure dans l'appendice A.

Une fois que tous les segments de la ligne k ont été identifiés, le t.t.c. est modifié de façon à refléter l'effet de ces nouvelles données. La section 3.1.1.1 donne suffisamment d'indications à ce sujet sans qu'il soit nécessaire d'insister davantage.

L'étape intitulée: "Remaniement du t.i.c." (fig. 9) consiste tout simplement à transférer dans la moitié gauche du tableau de la fig. 3 toute l'information contenue dans la moitié droite. Autrement dit, la ligne k devient la ligne k-1 et l'extracteur est prêt à traiter une nouvelle ligne de l'image.

La dernière étape du processus d'extraction consiste en une mise à jour finale du t.t.c.. Comme dans le cas d'une mise à jour partielle, on commence par regrouper les objets contigus pour ensuite supprimer les objets aberrants. En outre, tel qu'indiqué dans la section 3.1.1.1, on calcule alors les traits caractéristiques proprement dit de tous les objets que l'extracteur a détectés.

3.2 Etiquetage et propagation des pixels individuels - Extracteur no. 2

Ce second extracteur est structuré exactement de la même façon que le précédent tout en étant conceptuellement plus simple du fait que la procédure d'identification des pixels individuels (réf. 5) est plus immédiate que celle des points de contours. Les réf. 2 et 3 décrivent

Abcisses des pixels	Numéros des pixels	Abcisses des pixels	Numéros des pixels
1	2	3	4
1	0	1	0
2		2	
3		3	
.		.	
.		.	
.		.	
.		.	
N_c	0	N_c	0

FIGURE 10 - Le tableau d'identification des pixels individuels comporte 4 colonnes et un nombre de lignes égal au nombre de colonnes (N_c) de l'image. Comme pour le t.i.c., la moitié droite du tableau est le pendant de la moitié gauche. La première se rapporte à la ligne courante et la seconde à la ligne qui précède.

en détail le synthétiseur de silhouettes, quoiqu'en des termes différents de ceux employés ici. Cependant, contrairement à ce qui est dit dans la réf. 2, cet extracteur, tout comme le précédent, segmente l'image, identifie les objets produits et extrait les traits caractéristiques de ces derniers en l'espace d'un seul balayage de l'image.

La mémoire de cet extracteur est également constituée de 2 tableaux: un tableau d'identification des pixels individuels (fig. 10), ou t.i.p., et un tableau des traits caractéristiques identique à celui décrit dans la section 3.1.1.1. Le tableau servant à l'identification des pixels individuels comporte, en principe, 4 colonnes et un nombre de lignes égal au nombre de colonnes, N_C , de l'image (fig. 10). Nous disons en principe parce qu'il est bien évident qu'il n'est pas nécessaire de réserver un bloc de mémoire pour les colonnes 1 et 3 du tableau puisque les abscisses correspondent à l'ordre de succession des pixels sur une ligne de l'image. Autrement dit, on garde constamment en mémoire 2 lignes consécutives de l'image segmentée où les pixels appartenant au fond de la scène sont représentés par des zéros. Les colonnes 1 et 3 du t.i.p. auraient leur raison d'être si seuls les pixels-objets d'une ligne étaient conservés en mémoire. Outre le fait qu'une telle façon de procéder ne réduirait guère l'encombrement du t.i.p. en mémoire, elle compliquerait énormément la procédure d'identification des pixels. L'initialisation du t.i.p. se fait en remplissant de zéros la deuxième colonne du tableau. Par ailleurs, pour éviter les effets de bout, on s'assurera que la première et la dernière ligne de la quatrième colonne du t.i.p. renferment toujours un zéro. On force de fait le pourtour de l'image à se confondre avec l'arrière-plan.

Avant d'aborder la procédure d'identification des pixels, il faut décider du nombre, 4 ou 8, de voisins immédiats d'un pixel. Duda et Hart (réf. 6) suggèrent 4 pour les pixels-objets et 8 pour ceux de l'arrière-plan. Notre expérience nous incite plutôt à considérer comme adjacents deux pixels ayant au moins un sommet en commun (fig. 11).

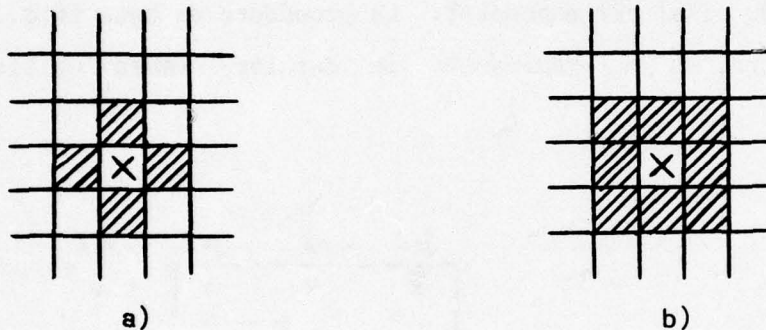


FIGURE 11 - Le pixel marqué d'un X peut être considéré comme ayant 4(a) ou 8(b) voisins immédiats.

L'hypothèse des 4 voisins est attrayante parce qu'elle réduit de façon sensible (cette conclusion ne s'applique pas à l'extracteur no. 1) le nombre moyen d'opérations par pixel. Par contre, à moins que la segmentation produise surtout des formes denses, il en résulte généralement un plus grand nombre d'objets, ce qui accroît le fardeau du classifieur. Il n'est donc pas facile de trancher la question. Dans ce cas comme dans bien d'autres, il faut considérer l'apport des opérations en amont et l'effet produit sur les opérations en aval. En d'autres termes, on ne peut évaluer une telle hypothèse de façon absolue mais uniquement en fonction d'un système donné pris dans son entier.

La présente procédure d'identification des objets de l'image segmentée n'est pas unique. On peut imaginer plusieurs variantes. Nous nous plaçons ici dans l'hypothèse des 8 voisins et nous supposons que l'extracteur conserve simultanément en mémoire tout au plus 2 lignes de l'image. Par ailleurs, nous partons de l'image segmentée, $\text{seg}(x,y)$, définie par l'éq. 1. Dans ce contexte, le problème consiste à remplacer les 1 par un numéro spécifique désigné $n_i(j)$, i et j étant les coordonnées du pixel correspondant. La procédure de base (fig. 12) est donc la suivante, où n_0 représente le dernier numéro utilisé pour étiqueter un pixel:

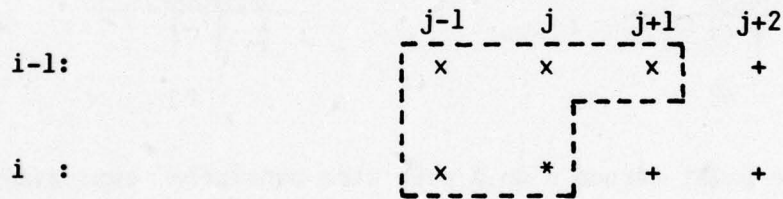


FIGURE 12 - Cette figure montre le pixel courant (i,j) et ses 4 voisins immédiats indiqués par un X.

- 1) si $n_{i-1}(j-1) \times n_{i-1}(j+1) \neq 0$
et $n_{i-1}(j-1) \neq n_{i-1}(j+1)$

on en déduit que les segments auxquels ces 2 pixels appartiennent sont reliés par la diagonale et, par conséquent, qu'ils se rattachent au même objet. On détermine lequel des 2 numéros est le plus grand, $n_{i-1}(j-1)$ disons, et on remplace, dans la table de substitution tous les $n_{i-1}(j-1)$ par $n_{i-1}(j+1)$.

- 2) Pose $n_i(j) = n_{i-1}(j-1)$ si $n_{i-1}(j-1) \neq 0$; sinon
 $n_i(j) = n_{i-1}(j)$ si $n_{i-1}(j) \neq 0$; sinon
 $n_i(j) = n_{i-1}(j+1)$ si $n_{i-1}(j+1) \neq 0$; sinon
 $n_i(j) = n_i(j-1)$ si $n_i(j-1) \neq 0$; sinon

- 3) on conclut que le pixel (i,j) n'est pas le prolongement d'un segment existant mais bien le début d'un nouveau segment et on pose:

$$n_i(j) = n_0 + 1.$$

- 4) Si $n_i(j-1) \neq 0$
 et $n_i(j-1) \neq n_i(j)$

nous nous trouvons en présence de deux segments reliés par une extrémité. Cette information est entrée dans la table de substitution de la manière décrite précédemment.

- 5) La procédure reprend en 1) avec le prochain pixel différent de zéro.

Une image segmentée comporte généralement plusieurs objets carrément aberrants, par exemple, en acquisition de cibles, les objets dont l'une des dimensions mesure moins de 3 pixels et, en particulier, ceux constitués d'un seul pixel. Ces objets surchargent inutilement l'extracteur et risquent même de le saturer. Il est par conséquent préférable de les éliminer sur-le-champ. Pour cela, il suffit de remplacer la précédente étape 3) par:

3') $n_i(j) = 0$ si $n_i(j+1) = 0$; sinon
 $n_i(j) = n_i(j+1) = n_{i-1}(j+2)$ si $n_{i-1}(j+2) \neq 0$; sinon
 $n_i(j) = n_i(j+1) = 0$ si $n_i(j+2) = 0$; sinon

on conclut que les pixels (i,j) et $(i,j+1)$ marquent le début d'un nouveau segment et on pose:

$$n_i(j) = n_i(j+1) = n_0 + 1.$$

Il est à noter qu'en procédant ainsi on élimine également certaines excroissances sur des objets dont la forme est autrement régulière.

Le nombre de lignes du tableau des traits caractéristiques (fig. 4) est forcément limité et, malgré les mises à jour répétées, il peut fort bien arriver que l'extracteur soit obligé de déclarer forfait parce que le tableau est plein. Pour réduire au minimum le nombre de mises à jour et pour éviter que le t.t.c. sature, on peut modifier la procédure de base de l'extracteur de telle sorte qu'il utilise le moins possible de numéros pour identifier les objets. Posons $j = j-1$ et soit $N(n_i(j))$ le nombre de pixels étiquetés $n_i(j)$. On remplace alors 4) par 4'):

4') si $n_i(j) \neq 0$ et $n_i(j) \neq n_i(j)$ pose

$$N(n_i(j)) = N(n_i(j)) + 1,$$

$$N(n_i(j)) = N(n_i(j)) - 1,$$

$$\text{et } n_i(j) = n_i(j);$$

de même si $n_i(j-1) \neq 0$ pose $j = j-1$ et on répète les 3 instructions précédentes; sinon

pose $n_o = n_o - 1$ si $N(n_i(j)) = 0$; sinon

on conclut qu'il existe sur la ligne précédente un segment étiqueté $n_i(j)$ et que ce segment appartient au même objet que le segment $n_i(j)$. On modifie donc en conséquence la table de substitution.

Il arrive fréquemment que cette simple procédure nous permette de récupérer un numéro.

Nous avons décrit dans cette section ce qui constitue l'essentiel du présent extracteur. Pour le reste, il se comporte grosso modo comme le précédent.

4.0 EVIDENCES EXPERIMENTALES

La fig. 13 montre une image binaire constituée d'un certain nombre de composantes connexes, ou objets, de forme arbitraire. Il pourrait s'agir d'une image produite par une opération quelconque de segmentation mais, en réalité, c'est une image synthétique destinée à illustrer le fonctionnement et les particularités des 2 extracteurs décrits dans la section précédente.

Une des quantités que l'on désire généralement connaître, c'est la taille des composantes connexes, définie comme étant le nombre de

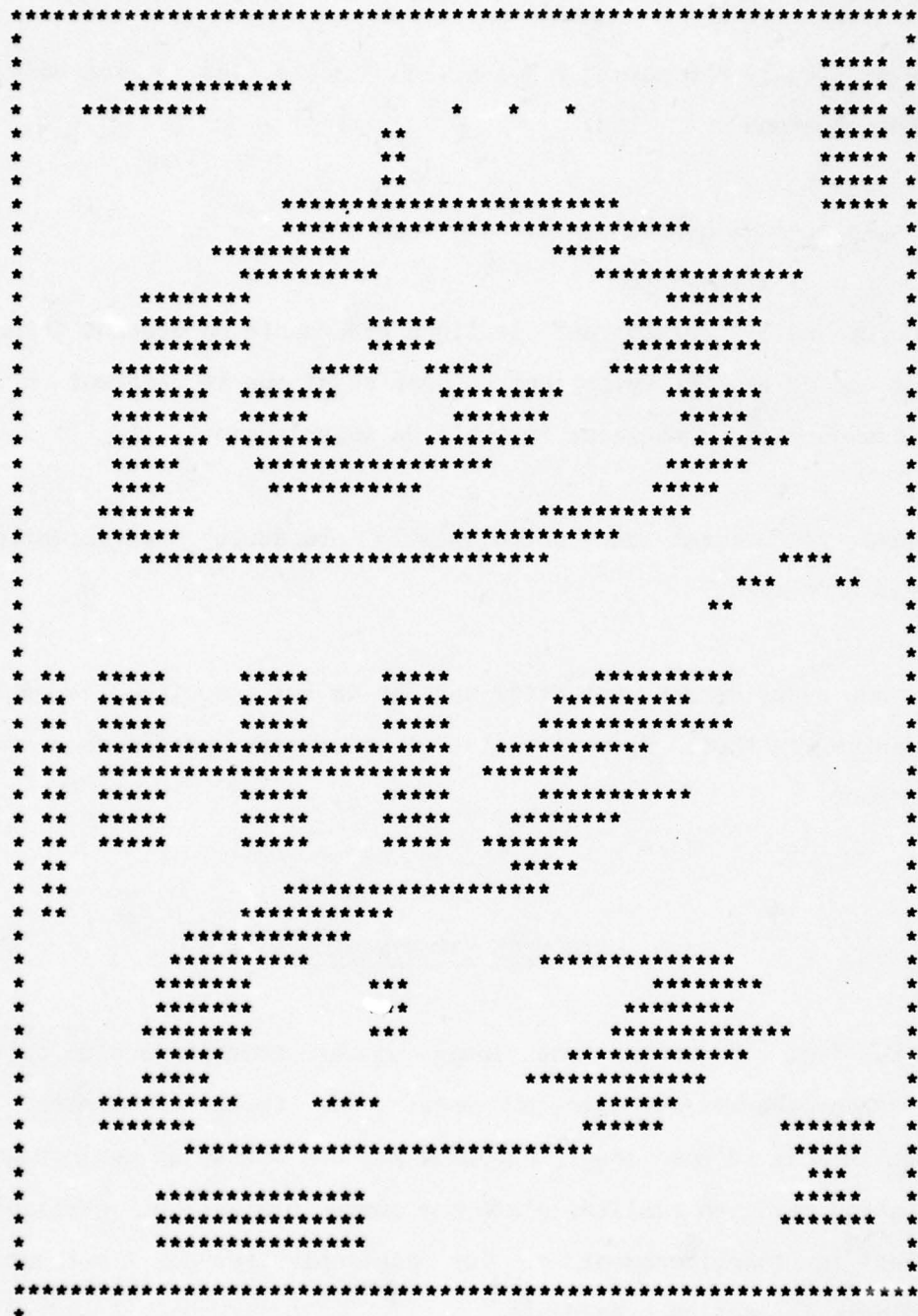


FIGURE 13 - Image binaire constituée d'un certain nombre de composantes connexes, ou objets, de forme arbitraire

pixels constitutifs. Pour cela, il faut d'abord délimiter les composantes et, par le fait même, les dénombrer. Dans le cas d'une image échantillonnée, comme ici, sur une grille rectangulaire, le nombre de composantes connexes peut varier selon l'hypothèse de connexité, quadruple ou octuple, retenue. Nous avons explicité, dans la section 3.2, les raisons pour lesquelles nous préférons l'hypothèse des 8 voisins (connexité octuple).

La nécessité d'une hypothèse de connexité ne s'impose pas avec autant d'évidence pour l'extracteur no. 1 que pour l'extracteur no. 2. Toutefois, si l'on se réfère à la section 3.1.2, on constatera que la procédure d'identification des points de contours recours implicitement à une connexité quadruple. C'est même la seule qu'elle admet. Pour en changer, il faut user d'un artifice qui consiste à allonger artificiellement les segments de la ligne k-1 (fig. 14) avant de procéder à l'identification des segments extraits de la ligne k. Il



FIGURE 14 - La fig. 14a est constituée de 3 objets si l'on retient l'hypothèse d'une connexité quadruple, mais d'un seul si l'on postule plutôt une connexité octuple. On peut toutefois obtenir le même résultat, sans changer d'hypothèse, en allongeant artificiellement, fig. 14b, le segment j.

TABLEAU I

Données numériques relatives à la figure 15

		15A		15B	15C		15D
NUMERO	SYMBOLE	TABLE DE SUBST.	NOMBRE DE PIXELS	NOMBRE DE PIXELS	TABLE DE SUBST.	NOMBRE DE PIXELS	NOMBRE DE PIXELS
1	1	1	35	35	1	35	35
2	2	2	21	21	2	21	284
3	3	3	1	1	3	632	92
4	4	4	1	1	3	348	11
5	5	5	1	1	5	118	140
6	6	6	638	290	6 ⁵	26	294
7	7	6	348	92	7	15	9
8	8	8	118	11	7	4	27
9	9	8	26	22	9	140	45
10	A	10	15	140	9	15	
11	B	10	4	294	9	15	
12	C	12	22	9	12	213	
13	D	13	140	5	9	15	
14	E	13	15	27	9	15	
15	F	13	15	45	12	81	
16	G	16	213		16	9	
17	H	13	15		17	5	
18	I	13	15		18	27	
19	J	16	81		19	45	
20	K	20	9				
21	L	21	5				
22	M	22	27				
23	N	23	45				

EOF HIT AFTER 39.

*

NON CLASSIFIE
39

```

*****
*
*
*      2*****2
*      2*****2      3  4  5
*
*          66
*          66
*          66
*          6*****6
*          6*****6
*          6*****7      7***6
*          6*****7      7*****6
*          6*****7      8***8      7*****6
*          6*****7      8***8      7*****6
*          6*****7      8***9      9***8      7*****6
*          6*****7      8*****9      9*****8      7*****6
*          6*****7      8***9      9*****8      7***6
*          6*****7      8*****9      9*****8      7*****6
*          6***7      8*****8      7***6
*          6**7      8*****8      7***6
*          6*****7      7*****6
*          6***7      7*****6
*          6*****7
*          6*****7
*          6*****6
*
*          A**A
*          A*B   BA
*          AA
*
*
*
*
* CC D***D      E***E      F***F      G*****G
* CC D***D      E***E      F***F      G*****G
* CC D***D      E***E      F***F      G*****G
* CC D*****D      G*****G
* CC D*****D      G*****G
* CC D***H      H***I      I***D      G*****G
* CC D***H      H***I      I***D      G*****G
* CC D***H      H***I      I***D      G*****G
* CC
* CC G*****G
* CC G*****G
*
*      G*****G
*      G*****G
*      G*****G      K*K
*      G*****G      K*K
*      G*****G      K*K
*      G***G
*      G***G
*      G*****G      L***L
*      G*****G
*      G*****G
*
*      N*****N
*      N*****N
*      N*****N
*
*****

```

FIGURE 15a - Résultat produit par la seule procédure d'identification
des points de contours de l'extracteur no. 1

NON CLASSIFIED
40

[illegible]

FIGURE 15b - Remaniement de la fig. 15a à l'aide de la table de substitution du tableau I

41

pixels de large.

42

★

FIGURE 15d - Remaniement de la fig. 15c à l'aide de la table de substitution du tableau I et élimination des objets mesurant moins de 3 pixels de haut: résultat final de l'extracteur no. 1

NON CLASSIFIE
43

TABLEAU II

Données numériques relatives à la figure 16

		16A		16B	16C		16D	
NUMERO	SYMBOLE	TABLE DE SUBST.	NOMBRE DE PIXELS	NOMBRE DE PIXELS	TABLE DE SUBST.	NOMBRE DE PIXELS	TABLE DE SUBST.	NOMBRE DE PIXELS
1	1	1	35	35	1	35	1	35
2	2	2	19	21	2	21	2	21
3	3	2	2	1	3	1	3	284
4	4	4	1	1	4	1	4	92
5	5	5	1	1	5	1	5	11
6	6	6	1	290	6	290	6	15
7	7	7	105	92	7	92	6	15
8	8	7	26	11	8	11	6	110
9	9	7	10	22	9	22	9	184
10	A	7	39	140	10	15	9	110
11	B	11	43	294	10	15	11	9
12	C	7	22	9	10	110	12	5
13	D	7	53	5	13	184	13	27
14	E	11	6	27	13	110	14	45
15	F	11	39	45	15	9		
16	G	11	4		16	5		
17	H	7	8		17	27		
18	I	7	15		18	45		
19	J	7	12					
20	K	20	7					
21	L	20	3					
22	M	20	1					
23	N	23	22					
24	O	24	55					
25	P	24	50					
26	Q	24	35					
27	R	27	31					
28	S	27	29					
29	T	27	15					
30	U	27	11					
31	V	27	1					
32	*	27	30					
33	1	27	10					
34	2	27	79					
35	3	27	47					
36	4	36	9					
37	5	27	20					
38	6	27	14					
39	7	27	2					
40	8	27	5					
41	9	41	5					
42	A	42	27					
43	B	43	45					

44

[illegible]

FIGURE 16a - Résultat produit par la seule procédure d'identification
des pixels individuels de l'extracteur no. 2

45

[illegible]

FIGURE 16b - Remaniement de la fig. 16a à l'aide de la table de substitution du tableau II

46

[illegible]

FIGURE 16c - Résultat produit par la seule procédure d'identification
des pixels individuels de l'extracteur no. 2 mais modifiée
de façon à réduire le nombre total de numéros

47

FIGURE 16d - Résultat final de l'extracteur no. 2

FIGURE 16d - Résultat final de l'extracteur no. 2

suffit pour cela de soustraire 1 aux abscisses de la colonne 5 du t.i.c. (fig. 3) et d'additionner 1 à celles de la colonne 6 avant de transférer toute la moitié droite du t.i.c. dans celle de gauche, au niveau du bloc intitulé: "Remaniement du t.i.c.", dans l'organigramme de la fig. 9.

Le tableau I rassemble les données numériques relatives à la fig. 15 et, par conséquent, à l'extracteur no. 1. Le tableau II fait de même pour la fig. 16 et l'extracteur no. 2. Les 2 premières colonnes de chacun de ces tableaux renferment respectivement l'ensemble des numéros employés pour étiqueter les composantes connexes et les symboles alphanumériques utilisés pour les représenter sur les figures correspondantes. Les autres colonnes des 2 tableaux s'expliquent d'elles-mêmes.

La figure 15a est le résultat de l'application de la procédure d'identification des points de contours à l'image de la fig. 13. On remarque que le nombre de numéros utilisés est plus grand que le nombre réel de composantes connexes. C'est généralement le cas avec l'un ou l'autre extracteur. Par ailleurs, le nombre de pixels (tableau I) des composantes 7 et 9 est négatif. Ainsi que nous l'avons souligné précédemment, ceci indique que ces numéros délimitent non pas un objet mais un trou. La table de substitution (tableau I) révèle que le trou 7 appartient à l'objet 6, et le trou 9 à l'objet 8. Remarquons que le nombre de pixels correspond toujours à un objet plein. Toutefois, comme l'étendue d'un trou est représentée par un nombre négatif, celui-ci se soustrait automatiquement lorsque vient le temps de regrouper les composantes partielles. Le nombre de pixels des composantes B, H et I (fig. 15a) est également négatif. Il s'agit ici de points creux et non

de trous. On pourrait facilement, si l'application considérée l'exigeait, modifier l'extracteur de telle sorte qu'il puisse distinguer un trou d'un point creux. Notons, au sujet de ces derniers, que l'extracteur voit uniquement les points creux qui pénètrent l'objet en sens inverse du balayage soit, dans le cas présent, du bas vers le haut. Ainsi, la composante représentée par la lettre A sur la fig. 15b possède 4 points creux mais l'extracteur n'en décèle que 2.

La table de substitution (tableau I) associée à la fig. 15a comprend uniquement 15 numéros distincts. C'est donc dire que l'image de la fig. 13 renferme au total 15 composantes connexes. Pour s'en convaincre et en même temps vérifier l'exactitude des résultats obtenus, il suffit de remanier la fig. 15a à l'aide de la table de substitution et ainsi obtenir la fig. 15b. Nous attirons l'attention sur la composante 8 de cette figure. C'est la preuve que l'hypothèse de connexité de l'extracteur est bien celle des 8 voisins puisque, dans le cas opposé, nous aurions 4 composantes au lieu de 1 car tous les segments ne sont reliés que par la diagonale.

La procédure de base de l'extracteur no. 2 produit beaucoup plus de composantes partielles (fig. 16a et tableau II), quoique le résultat final (fig. 16b) soit le même. Par contre, si l'on remplace l'étape 4) de la procédure de base par 4'), on constate effectivement (fig. 16c) une réduction appréciable du nombre de numéros employés. Cette constatation n'est nullement limitée à cet exemple précis mais est de portée générale.

Nous avons indiqué dans la section 3.2 qu'il était préférable d'éliminer sur-le-champ certaines composantes aberrantes et, en particulier, si l'application le permet, les composantes dont l'une des dimensions mesure moins de 3 pixels. Ainsi, en ce qui concerne l'extracteur no. 2, si l'on substitue l'étape 3') à l'étape 3) (voir section 3.2) on obtient l'image de la fig. 16d au lieu de celle de la fig. 16c. On notera que les composantes 3, 4, 5 et 9 de la fig. 16c ont été supprimées. Cependant, les composantes 2 et 6 de la même figure subsistent, même si elles mesurent moins de 3 pixels de haut. De façon générale, étant donné que l'extracteur ne conserve en mémoire que 2 lignes de l'image, il n'est pas possible d'éliminer ainsi les composantes mesurant moins de 3 pixels de haut mais plus de 3 pixels de large. Pour y parvenir il est précisément nécessaire de connaître la hauteur des composantes. En pratique, l'élimination de ces dernières composantes aberrantes se trouve reportée à la mise à jour la plus près, du tableau des traits caractéristiques. Par ailleurs, si l'on compare les fig. 16c et 16d, on se rend compte que la composante 6, fig. 16c, a été amputée d'une excroissance qui la surmontait. Encore une fois, comme l'extracteur ne voit simultanément que 2 lignes de l'image, il lui est impossible de faire la différence entre une telle excroissance et une composante aberrante comme la composante 9 (fig. 16c). Il est important de souligner, en outre, que les excroissances perpendiculaires à la direction de balayage ne sont nullement touchées ce qui représente un risque potentiel de distorsion.

La procédure d'élimination des composantes aberrantes en ce qui a trait à l'extracteur no. 1 est différente, mais le résultat (fig. 15c) est le même. Le résultat final, avec l'un ou l'autre extracteur, est tel qu'il apparaît à la figure 15d.

5.0 CONCLUSION

Nous avons décrit dans ce rapport 2 extracteurs séquentiels utilisés dans le cadre d'un système automatique d'acquisition de cibles sur images infrarouge. Toutefois, ces extracteurs ne sont pas en soi limités à une application particulière. On peut les mettre en oeuvre partout où il est nécessaire d'effectuer des mesures quantitatives sur des formes quelconques représentées sur une image. On suppose évidemment que lesdites formes peuvent être extraites de l'image, c'est-à-dire une opération préalable de segmentation, laquelle est fonction de la nature de l'image ainsi que de l'application considérée. On peut en quelque sorte assimiler la segmentation à une interface dont le rôle consiste à définir des objets qui vont alimenter l'extracteur.

On peut particulariser un objet, soit en traçant son pourtour, soit en identifiant tous ses points constitutants. Dans le premier cas, celui de l'extracteur no. 1, nous avons démontré qu'il suffit de 4 inégalités simples, et non de 8 comme dans la réf. 4, pour identifier complètement le contour d'un objet. Le second procédé, par contre, requiert un nombre de comparaisons qui est fonction du nombre de voisins immédiats que l'on associe à un point de l'image. Il est égal à 4 si l'on pose, au départ, qu'un point d'image possède 8 voisins immédiats, et à 2 si l'on ne considère que les 4 voisins situés à une distance unitaire du point en question. Le nombre moyen d'opérations par pixel de l'extracteur no. 2, contrairement à l'extracteur no. 1, dépend donc fortement de l'hypothèse de connexité retenue.

Au nombre des avantages de l'extracteur no. 1 vis-à-vis l'extracteur no. 2, il faut mentionner sa capacité de déterminer si un objet possède, ou non, un ou plusieurs trous. C'est une particularité qui peut être intéressante pour certaines applications. Par exemple, une vue de face, dans l'infrarouge, d'un transport de troupes blindé (réf. 2 et 3) montre une silhouette présentant une large ouverture rectangulaire en son centre. Cette caractéristique peut servir à le distinguer d'un char d'assaut, d'une jeep ou de tout autre véhicule.

Aucun des 2 extracteurs décrits dans le présent rapport ne semble posséder, de façon générale, un avantage écrasant sur l'autre. Une plus grande familiarisation et une expérimentation plus poussée avec l'un et l'autre fera peut-être ressortir, à la longue, des critères permettant de mieux les départager. Mais la question ne sera sans doute véritablement tranchée que lorsque viendra le temps de les réaliser en pratique.

6.0 REFERENCES

1. Sévigny, L., "Le traitement des images et l'acquisition d'objectif dans l'infrarouge", CRDV R-4050/76, avril 1976, NON CLASSIFIE
2. Sévigny, L., "Simulation d'un système d'acquisition automatique d'objectif infrarouge dans un contexte sol-sol", CRDV R-4081/77, juin 1977, NON CLASSIFIE
3. Sévigny, L., "La reconnaissance de forme et l'acquisition d'objectif en infrarouge: nouvel algorithme de détection", CRDV R-4099/78, mars 1978, NON CLASSIFIE
4. Agrawala, Ashok K. and Kulkarni, Ashok V., "A Sequential Approach to the Extraction of Shape Features", Computer Graphics and Image Processing 6, 538-577 (1977)
5. Rosenfeld, A. and Kak, A. C., "Digital Picture Processing", Academic Press, 1976
6. Duda, R. O. and Hart, P. E., "Pattern Classification and Scene Analysis", Wiley, 1973

APPENDICE A

Organigramme de la procédure d'identification des points de contours

La figure A-1 se trouvant à la fin de cet appendice donne l'organigramme complet de la procédure d'identification des points de contours de l'extracteur no. 1 (fig. 9). Les symboles utilisés et leurs relations avec les figures et le corps du texte sont définis et explicités ci-dessous:

- J[I]: indice désignant un segment particulier de la ligne k-1[k];
KP[KC]: nombre de segments de la ligne k-1[k];
NOB: numéro utilisé pour étiqueter un point de contour;
LPG(J): abscisse, $x_{Lj}(k-1)$, du p.c.g. du j ième segment de la ligne k-1 (colonne 1 du t.i.c.);
LPD(J): abscisse, $x_{Rj}(k-1)$, du p.c.d. du j ième segment de la ligne k-1 (colonne 2 du t.i.c.);
LCG(I): abscisse, $x_{Li}(k)$, du p.c.g. du i ième segment de la ligne k (colonne 5 du t.i.c.);
LCD(I): abscisse, $x_{Ri}(k)$, du p.c.d. du i ième segment de la ligne k (colonne 6 du t.i.c.);
LPNG(J): numéro d'identification du p.c.g. [p.c.d.] du j ième segment
[LPND(J)] de la ligne k-1 (colonnes 3 et 4 du t.i.c.);
LCNG(I): numéro d'identification du p.c.g. [p.c.d.] du i ième segment
[LCND(I)] de la ligne k (colonnes 7 et 8 du t.i.c.);
TBLS(): table de substitution (colonne 10 du t.t.c.);
NPIX(): nombre de pixels, $N(n_i(j))$, étiquetés $n_i(j)$.

NON CLASSIFIE
55

Le sous-programme TABLE sert à modifier la table de substitution de la façon décrite à la fin de la section 3.1.2 du rapport.

Les losanges en pointillé correspondent aux inégalités 20,22,25 et 27 du texte. Par ailleurs, la colonne d'instructions à l'extrême droite de l'organigramme représente la procédure utilisée pour rechercher l'objet auquel un trou appartient.

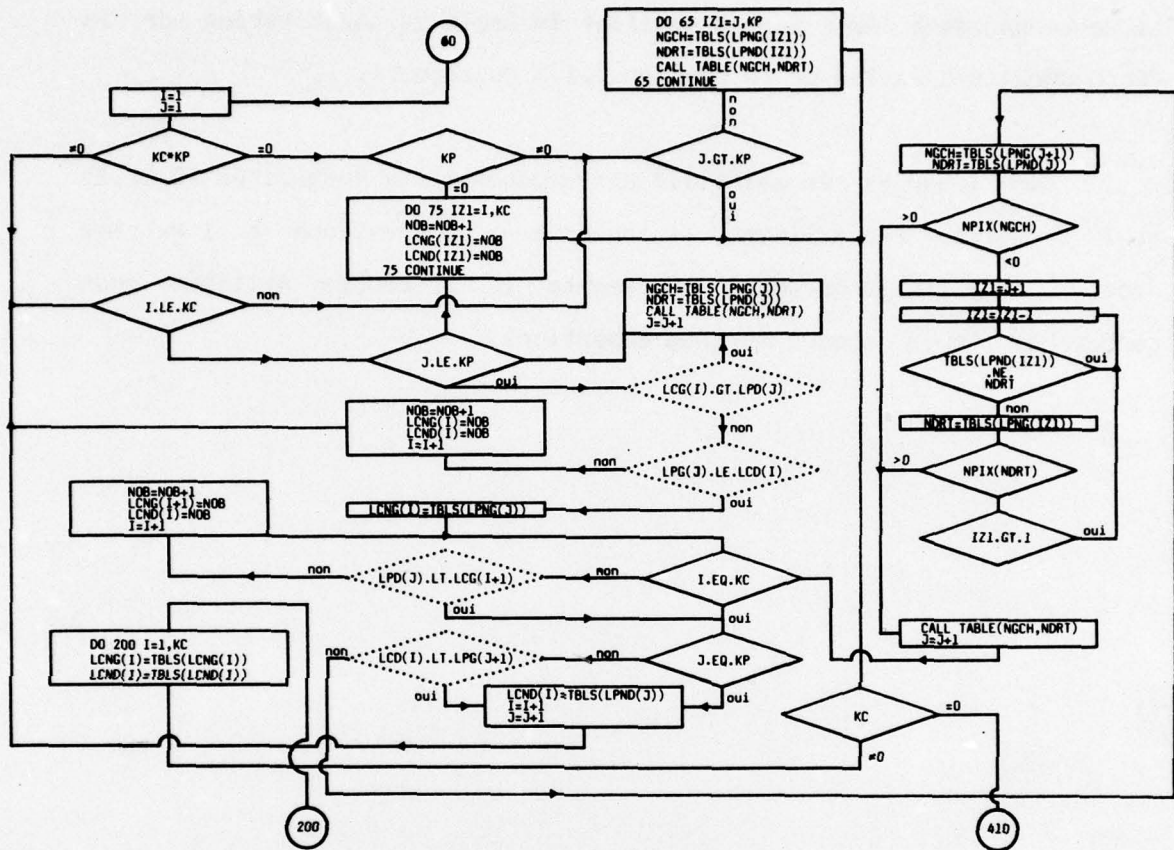


FIGURE A-1

CRDV R-4153/79 (NON CLASSIFIE)

Bureau - Recherche et Développement, MDN, Canada.
CRDV, C.P. 880, Courcellette, Qué. G0A 1R0

"Extracteurs séquentiels pour l'acquisition de cibles sur images"
par L. Sévigny

Ce rapport décrit 2 extracteurs séquentiels d'usage général bien qu'ils aient été développés en fonction d'une application bien particulière, à savoir l'acquisition de cibles sur images. L'extraction des traits caractéristiques implique une opération préalable visant à délimiter les composantes connexes, ou objets, représentés sur une image. Les 2 extracteurs réalisent cette opération par étiquetage et propagation; l'un, des points de contours de l'objet et l'autre des pixels constituant eux-mêmes. Dans le premier cas, nous démontrons qu'il suffit de 4 inégalités simples, et non de 8 comme dans l'article de Agrawala et Kulkarni, pour identifier complètement le contour d'un objet. Le second procédé, par contre, requiert un nombre de comparaisons qui est fonction du nombre de voisins immédiats que l'on associe à un point de l'image. Toutefois, aucun des 2 extracteurs ne semble posséder, globalement, un avantage écrasant sur l'autre. (NC)

CRDV R-4153/79 (NON CLASSIFIE)

Bureau - Recherche et Développement, MDN, Canada.
CRDV, C.P. 880, Courcellette, Qué. G0A 1R0

"Extracteurs séquentiels pour l'acquisition de cibles sur images"
par L. Sévigny

Ce rapport décrit 2 extracteurs séquentiels d'usage général bien qu'ils aient été développés en fonction d'une application bien particulière, à savoir l'acquisition de cibles sur images. L'extraction des traits caractéristiques implique une opération préalable visant à délimiter les composantes connexes, ou objets, représentés sur une image. Les 2 extracteurs réalisent cette opération par étiquetage et propagation; l'un, des points de contours de l'objet et l'autre des pixels constituant eux-mêmes. Dans le premier cas, nous démontrons qu'il suffit de 4 inégalités simples, et non de 8 comme dans l'article de Agrawala et Kulkarni, pour identifier complètement le contour d'un objet. Le second procédé, par contre, requiert un nombre de comparaisons qui est fonction du nombre de voisins immédiats que l'on associe à un point de l'image. Toutefois, aucun des 2 extracteurs ne semble posséder, globalement, un avantage écrasant sur l'autre. (NC)

CRDV R-4153/79 (NON CLASSIFIE)

Bureau - Recherche et Développement, MDN, Canada.
CRDV, C.P. 880, Courcellette, Qué. G0A 1R0

"Extracteurs séquentiels pour l'acquisition de cibles sur images"
par L. Sévigny

Ce rapport décrit 2 extracteurs séquentiels d'usage général bien qu'ils aient été développés en fonction d'une application bien particulière, à savoir l'acquisition de cibles sur images. L'extraction des traits caractéristiques implique une opération préalable visant à délimiter les composantes connexes, ou objets, représentés sur une image. Les 2 extracteurs réalisent cette opération par étiquetage et propagation; l'un, des points de contours de l'objet et l'autre des pixels constituant eux-mêmes. Dans le premier cas, nous démontrons qu'il suffit de 4 inégalités simples, et non de 8 comme dans l'article de Agrawala et Kulkarni, pour identifier complètement le contour d'un objet. Le second procédé, par contre, requiert un nombre de comparaisons qui est fonction du nombre de voisins immédiats que l'on associe à un point de l'image. Toutefois, aucun des 2 extracteurs ne semble posséder, globalement, un avantage écrasant sur l'autre. (NC)

DREV R-4153/79 (UNCLASSIFIED)

Research and Development Branch, DND, Canada.
DREV, P.O. Box 880, Courcellette, Que. G0A 1R0

"Sequential extractors for target acquisition on images"
by L. Sévigny

This report describes 2 general-purpose sequential extractors that were primarily developed for a specific application, namely target acquisition on images. The process of extracting features implies an initial operation aimed at delimitating the connected components, or objects, represented on an image. Both extractors realize this operation by labeling and propagating, for the first extractor, the boundary points and for the second one, the pixels themselves. In the first case, we show that one needs only 4 simple inequalities, instead of 8 as in the article by Agrawala and Kulkarri, to fully identify the boundaries of an object. The second process, on the contrary, needs a number of comparisons that is a function of the number of nearest neighbors one assigns to an image point. However, on the whole, none of the 2 extractors seem to possess an overwhelming advantage over the other. (U)

DREV R-4153/79 (UNCLASSIFIED)

Research and Development Branch, DND, Canada.
DREV, P.O. Box 880, Courcellette, Que. G0A 1R0

"Sequential extractors for target acquisition on images"
by L. Sévigny

This report describes 2 general-purpose sequential extractors that were primarily developed for a specific application, namely target acquisition on images. The process of extracting features implies an initial operation aimed at delimitating the connected components, or objects, represented on an image. Both extractors realize this operation by labeling and propagating, for the first extractor, the boundary points and for the second one, the pixels themselves. In the first case, we show that one needs only 4 simple inequalities, instead of 8 as in the article by Agrawala and Kulkarri, to fully identify the boundaries of an object. The second process, on the contrary, needs a number of comparisons that is a function of the number of nearest neighbors one assigns to an image point. However, on the whole, none of the 2 extractors seem to possess an overwhelming advantage over the other. (U)

DREV R-4153/79 (UNCLASSIFIED)

Research and Development Branch, DND, Canada.
DREV, P.O. Box 880, Courcellette, Que. G0A 1R0

"Sequential extractors for target acquisition on images"
by L. Sévigny

This report describes 2 general-purpose sequential extractors that were primarily developed for a specific application, namely target acquisition on images. The process of extracting features implies an initial operation aimed at delimitating the connected components, or objects, represented on an image. Both extractors realize this operation by labeling and propagating, for the first extractor, the boundary points and for the second one, the pixels themselves. In the first case, we show that one needs only 4 simple inequalities, instead of 8 as in the article by Agrawala and Kulkarri, to fully identify the boundaries of an object. The second process, on the contrary, needs a number of comparisons that is a function of the number of nearest neighbors one assigns to an image point. However, on the whole, none of the 2 extractors seem to possess an overwhelming advantage over the other. (U)